

## 5 Greedy Algorithm Proofs

1. **Backpacking planning:** You and some friends are planning a backpacking trip to the Sierras. You want to hike as far as possible each day but do not want to hike at night. On a map you have identified a large set of good camping sites. You plan to use the following system to decide where to camp each day. Each time you come to a one of the identified camp sites, you determine whether you can make it to the next good site before nightfall. If you can make it to the next site before nightfall, you will continue hiking otherwise you will stop.

Although there are drawbacks, e.g. you may miss staying in some great spots, you believe the system has one good feature, namely: “Given that we are only hiking in the day, it minimizes the number of camping stops we will have to make.”

You want to determine if this claim is true or false. To model this problem, you decide to make the following assumptions. Model the trail as a long straight path of length  $L$  and assume you and your friends can hike a maximum of  $d$  miles each day. Also you check the maps and determine that the campsites are located at distances  $x_1, x_2, x_3, \dots, x_n$  along the trail, with  $x_i < x_{i+1}$  for all  $i$ . Also, you believe that group will always correctly determine whether or not they can reach the next campsite by nightfall.

This means we can consider a set of stopping points **valid** if the distance between each adjacent pair is  $\leq d$ . Thus if you stop only at the points (campsites) in a **valid** set you will be able to make the full trip in daylight. You know it is possible to make the full trip in daylight since the full set of campsites is valid.

- A. Give pseudo code that describes the (greedy) algorithm that you and your friends will use to find a valid set of campsites with the fewest number of campsites.
  - B. Prove this algorithm always finds the optimal solution.
  - C. Analyze the algorithm’s complexity.
2. **Cell Tower Placement:** There is a long straight country road with expensive houses scattered along it. The houses are owned by affluent stock traders who require cell phone service. You consult for the company that needs to provide the cell service to every house without exception. The towers only have a range of four miles. You want to place the cell phone towers at locations along the road so that no house is more than four miles from the nearest cell phone tower. You know exact mileage along the road where each house is located. Design a greedy algorithm that will determine the set of locations for the cell towers that requires the fewest cell towers. Denote the location of the  $i^{\text{th}}$  house as  $M_i$  and the location of the  $j^{\text{th}}$  cell tower by  $T_j$ .
- A. Specify using pseudo code an efficient greedy algorithm to achieve this goal with the fewest cell towers.
  - B. Prove your algorithm always finds the optimal solution.
  - C. Analyze your algorithms complexity.

### 3. Triathlon Competitor Scheduling:

You are trying to schedule triathlon contestants so that the triathlon completes as early in the day as possible. (The triathlon completes when all the contestants are done.) Each contestant has a projected swimming time, running time, and biking time. Since the pool is very small only one contestant can be in the pool at a time. Any number of contestants can be biking and running at the same time. The contestants first swim, then bike, and finally run. Assuming that each contestant will complete each event in their projected time, what is the best order to send the people out, if you want the whole competition to be over as soon as possible? That is, give an efficient algorithm that produces a schedule whose completion time is as early as possible. **(10 points)**

- A. Specify using pseudo code an efficient greedy algorithm to achieve the goal of ending the competition at the earliest time.
- B. Prove your algorithm always finds the optimal solution.
- C. Analyze your algorithms complexity.

You must use the following notation:

**$S_i$ :  $i^{\text{th}}$  contestants swim time;**       **$B_i$ :  $i^{\text{th}}$  contestants bike time;**       **$R_i$ :  $i^{\text{th}}$  contestants run time;**

If your argument involves different orderings make sure your notation clearly differentiates the two orderings. Follow **the structure of the proof in class** used to prove correctness of the algorithm that solves Minimizing the Total Weighted Time to Completion. There will be slight differences since the function to be minimized here does not involve weights.

4. **Licensing strategy:** Suppose you own a company that must license software modules. Since your company has only a limited amount of money to spend each month, you can only purchase one license per month. The costs of the software licenses are all different and are given by  $p_1, \dots, p_n$ . Unfortunately the cost of all the licenses goes up by a factor of  $r$  ( $r$  is  $> 1$ ) each month,  $r$  is the same for all the products and constant over time. Thus the price of a license for the  $i^{\text{th}}$  product is  $p_i * r^m$  after  $m$  months.
  - A. Specify using pseudo code an efficient greedy algorithm to achieve this goal lowest total cost.
  - B. Prove your algorithm always finds the optimal solution.
  - C. Analyze your algorithms complexity.